

AD-A182 855

A SIMPLE WINDOW RANDOM ACCESS ALGORITHM WITH
ADVANTAGEOUS PROPERTIES. (U) VIRGINIA UNIV
CHARLOTTESVILLE DEPT OF ELECTRICAL ENGINEERING.

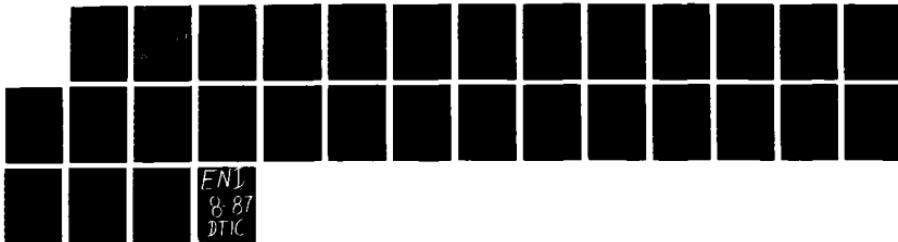
S/1

UNCLASSIFIED

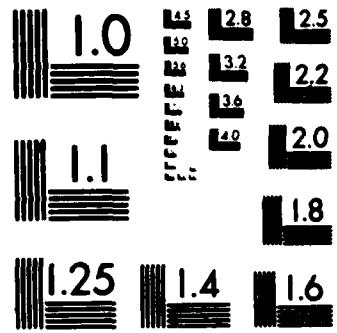
M PATERAKIS ET AL. JUL 87

F/B 12/2

NL



EN1
8-87
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

DTIC FILE COPY

(12)

Technical Report
Grant No. N00014-86-K-0742
September 1, 1986 - August 31, 1988

A SIMPLE WINDOW RANDOM ACCESS ALGORITHM
WITH ADVANTAGEOUS PROPERTIES

Submitted to:

Office of Naval Research
800 N. Quincy Street
Arlington, Virginia 22217-5000

Attention: Dr. Rabinder N. Madan
Electronics Division
Code 1114SE

Submitted by:

M. Paterakis
Graduate Research Assistant
P. Kazakos
Professor

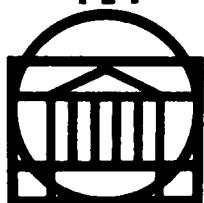
DTIC
SELECTED
S JUL 30 1987
C&D

Report No. UVA/525415/EE88/106

July 1987

DISTRIBUTION STATEMENT A

Approved for public release
Distribution Unlimited



SCHOOL OF ENGINEERING AND
APPLIED SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

UNIVERSITY OF VIRGINIA
CHARLOTTESVILLE, VIRGINIA 22901

87 7 27 039

Technical Report
Grant No. N00014-86-K-0742
September 1, 1986 - August 31, 1988

A SIMPLE WINDOW RANDOM ACCESS ALGORITHM
WITH ADVANTAGEOUS PROPERTIES

Submitted to:

Office of Naval Research
800 N. Quincy Street
Arlington, Virginia 22217-5000

Attention: Dr. Rabinder N. Madan
Electronics Division
Code 1114SE

Submitted by:

M. Paterakis
Graduate Research Assistant

P. Kazakos
Professor

Department of Electrical Engineering
SCHOOL OF ENGINEERING AND APPLIED SCIENCE
UNIVERSITY OF VIRGINIA
CHARLOTTESVILLE, VIRGINIA

A152115
REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION <u>Unclassified</u>	1b. RESTRICTIVE MARKINGS <u>None</u>	
2a. SECURITY CLASSIFICATION AUTHORITY	2c. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release, distribution unlimited	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE		
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UVA/525415/EE88/106	5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION University of Virginia Dept. of Electrical Engineering	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Office of Naval Research Resident Representative
6c. ADDRESS (City, State, and ZIP Code) Thornton Hall Charlottesville, VA 22901	7b. ADDRESS (City, State, and ZIP Code) Joseph Henry Building, Room 623 2100 Pennsylvania Ave., N. W. Washington, DC 20037-3202	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Dept. of the Navy Office of Naval Research	8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0742
8c. ADDRESS (City, State, and ZIP Code) 800 North Quincy Street Arlington, VA 22217-5000	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. PROJECT NO. TASK NO. WORK UNIT ACCESSION NO.	

11. TITLE (Include Security Classification)
A Simple Window Random Access Algorithm With Advantageous Properties

12 PERSONAL AUTHOR(S) M. Paterakis/P. Kazakos			
13a. TYPE OF REPORT Technical	13b. TIME COVERED FROM 9/1/86 TO 8/31/88	14. DATE OF REPORT (Year, Month, Day) 1987, July	15. PAGE COUNT

16. SUPPLEMENTARY NOTATION

17. COSATI CODES FIELD GROUP SUB-GROUP	18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

In this paper, a simple full sensing window random access algorithm is analyzed in the presence of the limit Poisson user model. The throughput of the algorithm is 0.43, and its delay and resistance to channel errors characteristics are superior to those induced by the Capetanakis window algorithm. In addition, the simple operations of the algorithm, in conjunction with its regenerative properties, allow for the computation and evaluation of the output traffic interdeparture distribution. The latter is needed in the evaluation of interacting systems which use the algorithm for their internal transmissions.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Rabinder N. Madan	22b. TELEPHONE (Include Area Code) (202) 696-4217
22c. OFFICE SYMBOL	

TABLE OF CONTENTS

	<u>Page</u>
I. Introduction	1
II. The System Model and the Algorithm	1
III. Throughput and Delay Analysis	2
IV. Performance in the Presence of Feedback Errors and Operations in Limited Sensing Environment	5
V. The Output Traffic Interdeparture Distribution	8
VI. Conclusions	10
References	11
Appendix	12

Accession For	
NTIS CRA&I <input checked="" type="checkbox"/> DTIC TAB <input type="checkbox"/> Unannounced <input type="checkbox"/>	
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail Ind or Special
A-1	



I. Introduction

In systems where independent users transmit through a single common channel, the deployment of random access transmission algorithms is frequently desirable, for the following reasons: (1) They are implemented independently by each user, without a prior coordination among the users. (2) They are insensitive to changing user population. (3) They induce low delays when the user traffic is bursty.

In this paper, we present and analyze a full sensing window random access algorithm. The algorithm was first proposed for systems with strict delay limitations, [3], and it requires that each user know the overall feedback history, (full sensing). As compared to other such existing algorithms, the present algorithm has the following interesting and beneficial properties: (1) It can be easily modified to operate in limited sensing environments, where each user follows the feedback history from the time he generates a message to the time when this message is successfully transmitted. (2) In the presence of the limit Poisson user model, the algorithm attains the same throughput as that attained by the Capetanakis' dynamic algorithm, [1], while it induces significantly lower delays for arrival rates above 0.30, and superior resistance to feedback errors. (3) The simple operations of the algorithm allow analysis and evaluation when strict delay limitations exist, [3]. Its simplicity, in conjunction with its regenerative properties, provide the means for the analysis and evaluation of the output traffic interdeparture distribution induced by the algorithm. The analysis of the latter distribution is important when several systems which use some Random Access Algorithm, (RAA), for internal transmissions interact, and it is not quite feasible when either the Capetanakis, [1], or the Gallager, [2], algorithms are deployed. (4) As compared to Gallager's algorithm, [2], the present algorithm operates in environments where the Poisson model is not valid, (e.g., when more than one packets can be generated within a given time instant), and can be then analyzed.

The organization of the paper is as follows: In section II, the system model is presented, and the algorithm is described. In section III, the throughput and delay analyses are included, in the presence of the limit Poisson user model, and in the absence of feedback errors. In section IV, the performance of the algorithm in the presence of feedback errors and its operations in limited sensing environments are discussed. In section V, the output traffic interdeparture distribution induced by the algorithm is analyzed and evaluated. In section VI, some conclusions are drawn.

II. The System Model and the Algorithm

We assume packet transmitting users, slotted channel, binary collision versus noncollision, (CNC), feedback per slot, no propagation delays, and absence of feedback errors. We also assume nonexistence of error correction coding; thus, collided packets are fully destroyed and retransmission is then necessary. Time is measured in slot units, slot t occupies the time interval $[t, t+1]$, and x_t denotes the feedback that corresponds to slot t ; $x_t = C$ and $x_t = NC$ represent then collision and noncollision slot t , respectively. For this system, let the following full sensing synchronous random access algorithm be deployed.

The algorithm utilizes a window of length Δ . Let t be a time instant such that, for some $t_1 < t$, all the packet arrivals in $(0, t_1]$ have been successfully transmitted by the algorithm and there is no information regarding the arrival interval $(t_1, t]$, and such that t corresponds to the

beginning of some slot. The instant t is then called Collision Resolution Point, (CRP), the arrival interval $(0, t_1]$ is called "resolved interval", and the interval $(t_1, t]$ is called "the lag at t ". In slot t , the packet arrivals in $(t_1, t_2 \leq \min(t_1 + \Delta, t)]$ attempt transmission, and the arrival interval $(t_1, t_2]$ is then called the "examined interval". If $(t_1, t_2]$ contains at most one packet, then it is resolved at t . If $(t_1, t_2]$ contains at least two packets, instead, then $x_t = C$, a collision occurs at t , and its resolution starts with slot $t+1$. Until the collision at t is resolved, no arrivals in (t_2, ∞) are allowed transmission. The time period required for the resolution of the latter collision is called the Collision Resolution Interval, (CRI). During some CRI, each user acts independently via the utilization of a counter whose value at time t is denoted r_t . The counter values can be either 1 or 2, and they are updated and utilized according to the rules below.

1. The user transmits in slot t , if and only if $r_t = 1$. A packet is successfully transmitted in t , if and only if $r_t = 1$ and $x_t = NC$.
2. The counter values transition in time as follows:
 - (a) If $x_{t-1} = NC$ and $r_{t-1} = 2$, then $r_t = 1$
 - (b) If $x_{t-1} = C$ and $r_{t-1} = 2$, then $r_t = 2$
 - (c) If $x_{t-1} = C$ and $r_{t-1} = 1$, then

$$r_t = \begin{cases} 1, & \text{with probability 0.5} \\ 2, & \text{with probability 0.5} \end{cases}$$

Remarks We note that the algorithmic operations can be depicted by a two-cell stack, where at each time instant t , cell 1 contains the transmitting users, (those with $r_t = 1$), and cell 2 contains the withholding users, (those with $r_t = 2$). The algorithm lumps, thus, the unsuccessful users together. In contrast, Capetanakis' algorithm distributes the unsuccessful users across the cells of an infinite-cell stack. As with Capetanakis' dynamic algorithm, the window size Δ is here subject to optimization for throughput maximization.

III. Throughput and Delay Analysis

In this section, we present the throughput and delay analyses of the algorithm, in the absence of feedback errors, in the full sensing environment, and in the presence of the limit Poisson user model. As proven in [5], the latter user model represents a lower bound. That is, when the user population is finite, the users are independent and identical, and the packet generation process per user is memoryless, then the throughput and delay characteristics of the algorithm are superior to those induced when the user environment is limit Poisson.

Let the system start operating at time zero, and let us consider the sequence in time of lags that are induced by the algorithm. Let C_i denote the length of the i -th lag, where $i \geq 1$. Then, the first lag corresponds to the empty slot zero; thus, $C_1 = 1$. In addition, the sequence $C_i; i \geq 1$ is a

Markov chain whose state space is at most countable. Let D_n denote the delay experienced by the n -th successfully transmitted packet arrival, as induced by the algorithm; that is, the time between the arrival of the packet and its successful transmission. Let the sequence T_i , $i \geq 1$ be defined as follows: Each T_i corresponds to the beginning of some slot, and $T_1 = 1$. Also, each T_i corresponds to the ending point of a length-one lag. T_{i+1} is then the ending point of the first after T_i unity length lag. Let R_i , $i \geq 1$ denote the number of successfully transmitted packets in the time interval $(0, T_i]$. Then, $Q_i = R_{i+1} - R_i$, $i \geq 1$ denotes the number of successfully transmitted packets in the interval $(T_i, T_{i+1}]$. The sequence Q_i , $i \geq 1$ is a sequence of i.i.d. random variables; thus R_i , $i \geq 1$ is a renewal process. In addition, the delay process D_n , $n \geq 1$ induced by the algorithm is regenerative with respect to the process R_i , $i \geq 1$ and the distribution of Q_i is nonperiodic, since $P(Q_i = 1) > 0$.

Let us define,

$$Z = E\{Q_1\}, \quad W = E\left\{\sum_{i=1}^{Q_1} D_i\right\} \quad (1)$$

From the regenerative arguments in [4], it follows that the expected per successfully transmitted packet steady-state delay, D , is given by the following expression:

$$D = W Z^{-1} \quad (2)$$

The effective computation of D relies on the successful derivation of upper and lower bounds on the quantities W and Z . Those bounds are found via the utilization of the methodology in [4], in conjunction with the quantities defined in [3]; the details are thus omitted. If $E\{I | \Delta, d\}$ denotes the expected length of a CRI, given that it starts with an examined interval of length Δ and with a lag d , then bounds on W and Z can be found only if:

$$\Delta > E\{I | \Delta, d\} \quad (3)$$

The inequality in (3) determines the stability region of the algorithm. Let us define,

$L_{n, k-n}$: The expected number of slots needed by the algorithm for the successful transmission of k packets, given that n of the k packets have counter values equal to 1 and that the remaining $k-n$ packets have counter values equal to 2.

Then, as found in [3], we have the following expressions, where λ denotes the intensity of the Poisson traffic process:

$$0 < L_{k,0} \leq \frac{3}{4} k^2 + \frac{9}{4} k - 2, \quad k \geq 1 \quad (4)$$

$$L_{0,0} = L_{1,0} = 1, \quad L_{0,i} = 1 + L_{i,0}, \quad i \geq 1 \quad (5)$$

$$E\{I | \Delta, d\} = \sum_{k=0}^{\infty} L_{k,0} e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} \quad (6)$$

The expressions in (4), (5), and (6), in conjunction with the methodology in [4], are used in the

computation of the algorithmic throughput and delays when the limit Poisson user model is present. In Table 1, we include the computed upper and lower bounds, D^u and D^l respectively, on the expected per packet delay D , for various Poisson intensities λ , and for both the present and the Capetanakis dynamic algorithms.

λ	Proposed algorithm		Capetanakis dynamic algorithm	
	D^l	D^u	D^l	D^u
0.02	1.562	1.563	1.563	1.564
0.06	1.708	1.716	1.713	1.719
0.10	1.888	1.917	1.903	1.921
0.16	2.257	2.363	2.308	2.362
0.20	2.607	2.812	2.712	2.809
0.24	3.103	3.467	3.308	3.476
0.30	4.412	5.197	4.976	5.365
0.32	5.162	6.170	5.973	6.501
0.36	7.941	9.665	9.798	10.883
0.38	11.008	13.398	14.121	15.855
0.40	18.262	22.024	24.427	27.736
0.42	57.354	67.665	78.530	90.212

Table 1

Upper and Lower Bounds on Steady-State Expected Delays

Regarding the throughput λ^* and the optimal window size Δ^* , the following results were found.

Proposed Algorithm:	$\lambda^* = 0.4295$	$\Delta^* = 2.33$
Capetanakis' Dynamic Algorithm:	$\lambda^* = 0.4295$	$\Delta^* = 2.673$

Table 2

Throughputs and Optimal Window Sizes

From Table 2, we observe that, in the presence of the limit Poisson user model, the algorithm in this paper attains the same throughput with the Capetanakis dynamic algorithm, but utilizes a smaller window size. From Table 1, we observe that the two algorithms induce practically identical delays for Poisson rates in $(0, 0.30)$, while for Poisson rates in $(0.30, 0.42]$, the present algorithm is significantly superior.

Remarks It may seem surprising that the algorithm in this paper attains the same throughput with the Capetanakis algorithm, and that it outperforms the latter in terms of delay performance. Indeed, the expected lengths $L_{k,0}$ in (4) are bounded by quadratic expressions, while the same lengths, L_k , in the Capetanakis algorithm are bounded by linear functions of k . However, $L_{2,0} = 4.5$ while $L_2 = 5$. At the same time, since $\Delta^* = 2.33$ for the present algorithm, the probability of a higher than two multiplicity collision is very small. The multiplicity-two events thus prevail, and the algorithm in this paper becomes superior to the Capetanakis' algorithm. We note that, as found in [3], the algorithm performs very well in environments where strict delay limitations exist. Then, it allows significant improvement in delay performance, at the expense of minimal loss in traffic. In addition, the analysis of the algorithmic performance when strict delay limitations exist is relatively simple, while the same analysis for the algorithms in [1] and [2] is then exceedingly complex.

IV. Performance in the Presence of Feedback Errors and Operations in the Limited Sensing Environment

Due to its simple rules, the present algorithm is much less sensitive to feedback errors than the Capetanakis' algorithm. To see that, let $\delta > 0$ be the probability that an empty slot is erroneously interpreted as a collision slot, and let this be the only form of feedback error. Then, as shown in [7], the throughput of the Capetanakis' dynamic algorithm reduces to zero, if δ exceeds 0.5. In contrast, the throughput λ_δ^* of the present algorithm remains positive, for every δ in $[0, 1]$. Indeed, given δ , let B_k^δ , E_k^δ , and S_k^δ respectively denote the expected number of collision, empty, and success slots during the resolution of a k -multiplicity collision, and let L_k^δ be the expected number of slots needed for the resolution of this collision. Then, the simple operations of the algorithm easily induce the following expressions:

$$B_k^\delta = B_k^0, \quad S_k^\delta = S_k^0, \quad E_k^\delta = (1-\delta)^{-1} E_k^0; \quad 0 \leq \delta < 1, \quad k \geq 2$$

$$L_k^\delta = B_k^0 + S_k^0 + E_k^0 (1-\delta)^{-1} = L_k^0 + \delta(1-\delta)^{-1} E_k^0 \leq (1-\delta)^{-1} L_k^0; 0 \leq \delta < 1, k \geq 2$$

$$L_0^\delta = (1-\delta)^{-2}; 0 \leq \delta < 1$$

$$\lambda_\delta^* = \sup_{x \geq 0} (x f_\delta^{-1}(x)) \quad (7)$$

where

$$f_\delta(x) \stackrel{\Delta}{=} \sum_{k=0}^{\infty} L_k^\delta e^{-x} \frac{x^k}{k!}, \quad L_k^\delta \stackrel{\Delta}{=} L_{k,0} \quad (8)$$

Thus,

$$\lambda_\delta^* \geq \lambda_0^* (1-\delta)^2 = (1-\delta)^2 (0.4295) > 0; \text{ for all } \delta \text{ in } [0,1) \quad (9)$$

Using methods as in [3], we developed bounds on the expected lengths L_k^δ , and then bounds on the throughput λ_δ^* , for $\delta \in (0, 1)$. We applied similar methods to compute bounds on the same throughput for the Capetanakis dynamic algorithm, (in [7] the Capetanakis nondynamic algorithm is considered). We exhibit our results in Figure 1. We observe the uniform superiority of the proposed algorithm. We notice in particular, that for $\delta = 0.5$, the proposed algorithm attains throughput 0.325, while the throughput of Capetanakis' dynamic algorithm is then zero.

We next considered the case when a single transmission may be interpreted as a collision, with probability ϵ , and when this is the only type of feedback error. Using the same methodology as above, we then computed bounds on the induced throughput, λ_ϵ^* , for both the proposed and the Capetanakis dynamic algorithms. Our results are included in Table 3. We observe that the two algorithms have practically identical performance in this case.

ϵ	Proposed algorithm λ_e^*	Capetanakis dynamic algorithm λ_e^*
0.02	0.420	0.422
0.04	0.412	0.414
0.06	0.403	0.407
0.08	0.395	0.399
0.10	0.387	0.392
0.20	0.346	0.353
0.30	0.305	0.313
0.40	0.265	0.273
0.50	0.225	0.231
0.60	0.184	0.187
0.70	0.142	0.142
0.80	0.098	0.097
0.90	0.052	0.049
1.00	0.000	0.000

Table 3

Throughput when with probability ϵ , a single transmission is interpreted as a collision.

Operations in the Limited Sensing Environment

In the limited sensing environment, it is required that each user monitor the channel feedback only from the time he generates a packet, to the time this packet is successfully transmitted. Therefore, the users' knowledge of the channel feedback history is then asynchronous. The objective in this case is to prevent new arrivals from interfering with some collision resolution in progress. This is possible, if each user can decide whether a collision resolution is in progress or not, within a finite number of slots from the time he generates a new packet. We observe that a user who has a new packet and observes a C slot decides to wait, since he can then deduce that there is some collision resolution in progress. Also, since a CRI ends with two consecutive NC slots, all the users who observe such an event, decide that there is no collision resolution in progress. In view of the above observations, we conclude that in the limited sensing environment, the algorithm can be modified to operate as follows:

The window size is the same as in the full sensing case. The window slides from present to past, however. In particular, the edge of the window is maintained one slot before the present time, and the window slides through the unexamined interval from present to past, (see Fig. 2). Within each window, the operations of the algorithm are the same as in the full sensing environment.

In the limited sensing environment, and for very light input traffic, the algorithm induces expected per packet delay equal to 2.5. As the rate of the input traffic increases, the expected

delays approach those induced in the full sensing environment. The throughput of the algorithm remains identical to that in the full sensing environment. In Figure 3, we plot the expected delays that the algorithm induces, in both the full sensing and the limited sensing environments. In the latter environment, the expected delays were computed via methodologies as those in [7]–[9].

Remarks We point out that the modification of Capetanakis' dynamic algorithm, to operate in the limited environment, is still an open and complex problem. In contrast, the modification is simple when the proposed algorithm is adopted. In systems where the Poisson user model is valid, the Part-and-Try algorithm with binary feedback is known to not operate in the limited sensing environment. [9], [10]. The throughput of the latter algorithm is then 0.45. But when the Poisson user model is not valid, it leads to deadlocks. In contrast, the proposed algorithm performs well in nonPoisson user systems.

V. The Output Traffic Interdeparture Distribution

In this section, we concentrate on the computation of the output traffic interdeparture distribution. In particular, we find analytically the steady-state distribution of the distance between two consecutive, successful transmissions, when the algorithm in this paper is adopted. We point out that the algorithm generates an output traffic process with memory. Our computations correspond thus to the first order distribution from this process. This first order distribution, in conjunction with a memoryless assumption, can be used as an approximation of the actual output traffic process, when studies of systems which deploy the algorithm and interact with each other are undertaken. Such interactions may correspond, for example, to: (1) Servicing the output traffic from several systems that deploy the algorithm, by a single server queue. (2) Transmitting the output traffic from a system that deploys the algorithm, through the transmission channel, of another random access system, (multi-hop problem). The methodology we use to compute the first order distribution from the output traffic process, extends easily to higher order distributions from the process as well. The computations become then exceedingly complex, however.

Our methodology utilizes the regenerative character of the output traffic process that the algorithm generates, and its steps are as those in [4]. The initial challenge here lies in the determination of regenerative points, which are pertinent to the output traffic process. We define the sequence $\{P_i\}_{i \geq 1}$ of such points as follows: Each P_i is a collision resolution point, (CRP), which follows a slot containing a successful transmission and at which the lag equals one. P_1 is the first after zero such CRP, and for every $i \geq 1$, P_{i+1} is the first after P_i such CRP. Let S_i , $i \geq 1$ denote the number of successful transmissions in $(0, P_i]$, and let d_n denote the distance between the $(n-1)$ -th and the n -th successful transmission. Then, S_i , $i \geq 1$ is a renewal process, and, as it can be easily seen, the process d_n , $n \geq 1$ is regenerative with respect to it. Let us define, $C_i = S_{i+1} - S_i$, $i \geq 1$. Then C_i denotes the number of successful transmissions in the internal $(P_i, P_{i+1}]$, where this interval will be called the i-th cycle. Let us define,

$$I_n(s) = \begin{cases} 1 & \text{if } d_n = s \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

$$H = E\{P_{i+1} - P_i\} \quad (11)$$

From the regenerative theorem [4], we then conclude that if $C = E\{C_1\} < \infty$, then,

$$\lim_{N \rightarrow \infty} N^{-1} \sum_{n=1}^N I_n(s) = \lim_{N \rightarrow \infty} E\left\{ \sum_{n=1}^N I_n(s) \right\} = C^{-1} E\left\{ \sum_{n=1}^{C_1} I_n(s) \right\} \quad (12)$$

; where if the intensity of the input Poisson traffic is λ , then,

$$C = \lambda H \quad (13)$$

In addition, since $P(C_1 = 1) > 0$, the distribution of C_1 is aperiodic and there exists a random variable d_∞ , such that the sequence $d_n, n=1,2,\dots$ converges in distribution to d_∞ . Then, d_∞ represents the steady state interdeparture distance induced by the algorithm, and its distribution satisfies the equality,

$$P(d_\infty = s) = C^{-1} E\left\{ \sum_{n=1}^{C_1} I_n(s) \right\} \quad (14)$$

The finiteness and the computation of the quantities C and $E\left\{ \sum_{n=1}^{C_1} I_n(s) \right\}$ in (14) are related to the existence and computation of appropriate solutions to infinite-dimensionality linear systems. Those systems and their solutions are included in the Appendix. In Table 4, we include the computed upper and lower bounds, respectively denoted P_s^u and P_s^l , on the probability $P(d_\infty = s)$, for input traffic Poisson intensities $\lambda = 0.1$ and $\lambda = 0.4$. In Figure 4, we plot the lower bounds against s , for various input traffic Poisson intensities λ .

s	$\lambda = 0.1$		$\lambda = 0.4$	
	P_s^l	P_s^u	P_s^l	P_s^u
1	0.1420	0.1427	0.4702	0.4728
2	0.0816	0.0832	0.2000	0.2048
3	0.0704	0.0739	0.0998	0.10806
4	0.0641	0.0696	0.0612	0.0787
5	0.0537	0.0603	0.0401	0.0603
6	0.0502	0.0591	0.0280	0.0397
7	0.0420	0.0503	0.0196	0.0264
8	0.0364	0.0452	0.0099	0.0173
9	0.0332	0.0431	0.0057	0.0094
10	0.0265	0.0393	0.0013	0.0062

Table 4

Upper and lower bounds on the interdeparture distribution for Poisson rates

$$\lambda = 0.1 \text{ and } \lambda = 0.4$$

From the results in Table 4 and Figure 4, we draw the following conclusions: (1) For low rates λ of the Poisson input traffic, ($\lambda \leq 0.1$), the interdeparture distribution is close to the Bernoulli distribution whose parameter is $p = \lambda e^{-\lambda}$. In particular, denoting $P_s = P(d_{\infty} = s)$, we then have $P_s \approx p(1-p)^{s-1}$, for $s \geq 2$. The probability P_1 , however, is then significantly larger than the Bernoulli parameter p . The intuitive explanation of the latter phenomenon goes as follows: For small rates λ , single arrivals in two consecutive slots occur with probability $p^2 = (\lambda e^{-\lambda})^2 = \lambda^2$, while the probability of a collision slot is then approximately equal to $2^{-1} \lambda^2 e^{-\lambda} = 2^{-1} \lambda^2$. Thus, for small rates λ , single arrivals in two consecutive slots contribute one third of P_1 , while the remaining two thirds are due to consecutive departures at the end of a collision resolution interval. (2) As the rate λ of the Poisson input traffic increases, the interdeparture distribution induced by the algorithm deviates further from the Bernoulli distribution. In fact, as λ increases, the mass of the interdeparture distribution accumulates at relatively small s values. For example, for $\lambda = 0.4$, we have $P_1 = 0.471$ and $\sum_{s=1}^{10} P_s - 1$.

Remarks Our results showed that it is generally wrong to conjecture exponential interdeparture distribution, (whose disscript form is Bernoulli). In fact, this distribution is far from exponential. Even for small input Poisson rates λ , the probability P_1 does not match the exponential fitting. We point out that our general approach in this section and the corresponding regenerative points apply to other algorithms as well, including the Capetanakis dynamic algorithm. However, the development of the appropriate recursions is then an exceedingly complex task. The simple operations of the proposed algorithm present a remarkable advantage, which does not characterize other existing algorithms.

VI. Conclusions

We presented a simple window random access algorithm for systems with binary, collision versus noncollision, feedback. We analyzed the algorithm in the presence of the limit Poisson user model, and for both its full sensing and limited sensing implementations. In addition to the throughput and the delay analyses, we studied the effect of feedback errors on the throughput of the algorithm and the output traffic interdeparture distribution, both in the full sensing environment. As compared to the Capetanakis dynamic algorithm, the proposed algorithm is superior in terms of delays and insensitivity to feedback errors. In contrast to the former, the algorithm can also be easily adapted for implementation in the limited sensing environment, it allows for analytical studying of the output traffic interdeparture distribution, and can be easily analyzed when strict delay limitations are imposed, [3].

References

- [1] J. I. Capetanakis, "Tree Algorithms for Packet Broadcast Channel, IEEE Trans. Inform. Theory, vol. IT-25, pp. 505-515, Sept. 1979.
- [2] R. G. Gallager, "Conflict Resolution in Random Access Broadcast Networks", in Proc. APOSOR Workshop Communication Theory and Applications, Provincetown, MA, Sept. 1978, pp. 74-76.
- [3] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "A Full Sensing Window Random-Access Algorithm for Messages with Strict Delay Constraints", Report No. UVA/525415/EE87/103, School of Engineering and Applied Science, University of Virginia, Charlottesville, Virginia, Feb. 1987. Also, submitted for publication.
- [4] L. Georgiadis, L. Merakos, and P. Papantoni-Kazakos, "A Method for the Delay Analysis of Random Multiple Access Algorithms whose Delay Process is Regenerative", IEEE Journal on Selected Areas in Communications, July 1987, to appear.
- [5] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "On the Relation between the Finite and the Infinite Population Models for a class of RAAs", IEEE Transactions on Communications, to appear.
- [6] S. Stidham, Jr., "Regenerative Processes in the Theory of Queues with applications to the Alternating-Priority Queue", Adv. Appl. Prob., Vol. 4, pp. 542-577, 1972.
- [7] J. L. Massey, "Collision Resolution Algorithms and Random Access Communications", in Multi-User Communication, Ed. G. Longo, Springer-Verlag, CISM.
- [8] P. Studer and H. Pletscher, "Q-ary Part-and-Try Algorithm for Packet Conflict Resolution", Inst. Telecommun. Swiss Fed. Inst. Technol., Zurich, Switzerland, Oct. 11, 1984.
- [9] L. Georgiadis and P. Papantoni-Kazakos, "A 0.487 Throughput Limited Sensing Algorithm", IEEE Trans. on Information Theory, Vol. IT-33, No. 2, March 1987.
- [10] P. Humbert, "On the Throughput of Channel Access Algorithms with Limited Sensing", IEEE Trans. on Communications, Vol. COM-34, April 1986, pp 345-347.

Appendix

We first provide some definitions.

- $l_{k,m}$: Given k packets with counter values equal to 1 and m packets with counter values equal to 2, the number of slots needed by the algorithm until the first successful transmission, (and including it), after the k -multiplicity collision has been observed.
- $n_{k,s}$: Given a collision resolution interval which starts with a k -multiplicity collision, the number of length s interdeparture intervals within it. The length from the initial collision to the first successful transmission is included in the counting.
- h_d : Starting with a CRP at which the lag equals d , $d \geq 1$, and which follows a successful transmission, the number of slots needed by the algorithm to reach the first lag-one CRP which follows a slot containing a successful transmission.
- $m_{d,s}$: Starting with a CRP at which the lag equals d , $d \geq 1$, and which follows a successful transmission, the number of length s interdeparture intervals until the first lag-one CRP which follows a slot containing a successful transmission. The distance from the initial CRP to the first successful transmission is included in the counting.
- $P(k,l,\delta | d)$: Given an arrival interval of length d , the probability that there are k arrivals in it, that $l_{k,0}=\delta$, and that it takes l slots for its resolution, including the initial collision slot.
- $P_k(l)$: Given a k -multiplicity initial collision, the probability that it takes l slots for its resolution, including the initial collision slot.

The above definitions are needed for the derivation of recursions that are pertinent to the infinite-dimensionality systems associated with the quantities in (14). We first note that:

$$H = E\{h_1\}, C = \lambda H \quad (A.1)$$

$$E\left\{ \sum_{n=1}^{C_1} I_n(s) \right\} = E\{m_{1,s}\} \quad (A.2)$$

Auxiliary Recursions

The operations of the algorithm induce the following recursions:

$$(I) \quad l_{1,m} = 0; \forall m, P(l_{k,m}=0) = 0; \forall k \geq 2, \forall m$$

$$l_{0,m} = 1 + l_{m,0}, P(l_{0,m}=1) = \begin{cases} 1, & \text{if } m=1 \\ 0, & \text{if } m \neq 1 \end{cases}$$

$$l_{k,m} = 1 + l_{i,m+k-i} ; \text{ with probability } \binom{k}{i} 2^{-k}, k \geq 2$$

$$P(l_{k,m} = s) = \begin{cases} 1, & \text{if } k=1 \text{ and } s=0 \\ P(l_{m,0} = s-1), & \text{if } k=0, s \geq 1 \\ 2^{-k} \sum_{i=0}^k \binom{k}{i} P(l_{i,m+k-i} = s-1), & \text{if } k \geq 2, s \geq 1 \end{cases}$$

(II)

$$n_{1,s} = \begin{cases} 1, & \text{if } s=1 \\ 0, & \text{if } s \neq 1 \end{cases}$$

$$k \geq 2 ; n_{k,s} = \begin{cases} n_{k-1,s} , & \text{with probability } P(l_{k,0} \neq s-1) \\ 1 + n_{k-1,s} , & \text{with probability } P(l_{k,0} = s-1) \end{cases}$$

$$N_{k,s} \stackrel{\Delta}{=} E\{n_{k,s}\} = \begin{cases} \sum_{i=2}^k P(l_{i,0} = s-1) , & \text{if } s > 1 \\ 1 + \sum_{i=2}^k P(l_{i,0} = 0) = 1, & \text{if } s = 1 \end{cases}$$

(III)

$$k=0,1 ; P_k(l) = \begin{cases} 1, & \text{if } l = 1 \\ 0, & \text{otherwise} \end{cases} , P_2(l) = P(l_{2,0} = l-2), \text{ for } l \geq 3$$

$$\left. \begin{array}{l} k \geq 2 \\ l \geq k+2 \end{array} \right\} ; P_k(l) = \sum_{s=1}^{l-k-1} P(l_{k,0} = s) P_{k-1}(l-s-1)$$

Given Poisson intensity λ ,

$$P(k,l,\rho | d) = e^{-\lambda d} \frac{(\lambda d)^k}{k!} P(l_{k,0} = \rho) P_{k-1}(l-\rho-1)$$

Recursions for h_d

Given Poisson intensity λ , and from the operations of the algorithm, we easily conclude:

$$d \leq \Delta ; h_d = \begin{cases} 1 , \text{ with probability } \lambda d e^{-\lambda d} \\ 1+h_1 , \text{ with probability } e^{-\lambda d} \\ l+h_l , \text{ with probability } \sum_{k=2}^{\infty} e^{-\lambda d} \frac{(\lambda d)^k}{k!} P_k(l), l \geq 2 \end{cases}$$

$$d > \Delta ; h_d = l+h_{d-\Delta+l} , \text{ with probability } \sum_{k=0}^{\infty} e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} P_k(l)$$

and thus,

$$\begin{aligned} H_d &\stackrel{\Delta}{=} E(h_d) = e^{-\lambda d} + E(l|d) + e^{-\lambda d} H_1 + \sum_{k \geq 2} \sum_{l \geq 1} e^{-\lambda d} \frac{(\lambda d)^k}{k!} P_k(l) H_l ; \quad d \leq \Delta \\ H_d &= E(l|\Delta) + \sum_{k \geq 0} \sum_{l \geq 1} e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} P_k(l) H_{d-\Delta+l} ; \quad d > \Delta \end{aligned} \quad (\text{A.1})$$

where,

$$E(l|d) = \sum_{k \geq 0} \sum_{l \geq 1} e^{-\lambda d} \frac{(\lambda d)^k}{k!} P_k(l).l$$

Recursions for m_{ds}

For $\lfloor \cdot \rfloor$ denoting integer part, for w.p. meaning with probability, and for Poisson intensiy λ , we conclude:

$$d \leq \Delta; m_{d,s} = \begin{cases} n_{1,s}; \text{ w.p. } e^{-\lambda d} \lambda d \\ n_{k,s} + m_{l,s}; \text{ w.p. } e^{-\lambda d} \frac{(\lambda d)^k}{k!} P_k(l); k \geq 2 \\ n_{k-1,s} + m_{l,s}; \text{ w.p. } P(0|d) \sum_{\substack{n \geq 0 \\ n+p+2 \neq s}} P^n(0|1) P(k,l,p|1); k \geq 2 \\ 1 + n_{k-1,s} + m_{l,s}; \text{ w.p. } e^{-\lambda(d+1)} \frac{\lambda^k}{k!} \sum_{\substack{n \geq 0 \\ n+p+2=s}} e^{-\lambda n} P(l_{k,0}=p) P_{k-1}(l-p-1); k \geq 2 \\ 1; \text{ w.p. } \lambda e^{-\lambda(d+s-1)}; s \geq 2 \end{cases}$$

For $d > \Delta$:

$$\begin{aligned} m_{d,s} &= n_{k,s} + m_{d-\Delta+l,s}; \text{ w.p. } e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} P_k(l); k \geq 1 \\ &= n_{k-1,s} + m_{d-n(\Delta-1)+l,s}; \text{ w.p. } \sum_{\substack{p=s-n-1 \\ 1 \leq n \leq \lfloor \frac{d-\Delta}{\Delta-1} \rfloor}} e^{-\lambda \Delta(n+1)} \frac{(\lambda \Delta)^k}{k!} P(l_{k,0}=p) P_{k-1}(l-p-1), k \geq 1, \text{ if } \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \geq 1 \\ &= 1 + n_{k-1,s} + m_{d-n(\Delta-1)+l,s}; \text{ w.p. } \sum_{\substack{1 \leq n \leq \lfloor \frac{d-\Delta}{\Delta-1} \rfloor}} e^{-\lambda \Delta(n+1)} \frac{(\lambda \Delta)^k}{k!} P(l_{k,0}=s-n-1) P_{k-1}(l-s+n), k \geq 1, \\ &\quad ; \text{ if } \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \geq 1 \\ n_{k-1,s} + m_{l,s}; \text{ w.p. } e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right]} \frac{\left[\lambda \left(d - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor (\Delta-1) \right) \right]^k}{k!} &\sum_{\lfloor \frac{d-\Delta}{\Delta-1} \rfloor + p+1 \neq s} P(l_{k,0}=p) P_{k-1}(l-p-1); k \geq 2 \\ = 1 + n_{k-1,s} + m_{l,s}; \text{ w.p. } e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right]} \frac{\left[\lambda \left(d - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor (\Delta-1) \right) \right]^k}{k!} P(l_{k,0}=s-1-\lfloor \frac{d-\Delta}{\Delta-1} \rfloor). &. P_{k-1} \left(l-s+\lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right); k \geq 2 \end{aligned}$$

$$\begin{aligned}
& n_{k-1,s} + m_{l,s} ; \text{ w.p. } e^{-\lambda(d+\lfloor \frac{d-\Delta}{\Delta-1} \rfloor + 1)} \sum_{n \geq 0} e^{-\lambda n} \frac{\lambda^k}{k!} P(l_{k,0} = \rho) P_{k-1}(l - \rho - 1); k \geq 2 \\
& \quad n + 2 + \rho + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \neq s \\
& = 1 + n_{k-1,s} + m_{l,s} ; \text{ w.p. } e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor + 1 \right]} \sum_{n \geq 0} e^{-\lambda n} \frac{\lambda^k}{k!} P \left(l_{k,0} = s - n - 2 - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right) . \\
& \quad . P_{k-1} \left(l + n - s + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor + 1 \right); k \geq 2 \\
& = 1 ; \text{ w.p. } \lambda e^{-\lambda(d+s-1)} ; \text{ if } s - 2 - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \geq 0 \\
& = 1 ; \text{ w.p. } \lambda \left[d - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor (\Delta-1) \right] e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right]} ; \text{ if } s = 1 + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor
\end{aligned}$$

Let us define,

$$U(x) \stackrel{\Delta}{=} \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

$$P_\delta(l) \stackrel{\Delta}{=} \sum_{k \geq 1} e^{-\delta} \frac{\delta^k}{k!} P_k(l)$$

$$N_{\delta,s} \stackrel{\Delta}{=} \sum_{k \geq 1} N_{k,s} e^{-\delta} \frac{\delta^k}{k!}, \quad N_{\delta,1} = 1 - e^{-\delta}$$

(A.2)

$$P_{\delta,\rho} \stackrel{\Delta}{=} \sum_{k \geq 1} e^{-\delta} \frac{\delta^k}{k!} P(l_{k,0} = \delta)$$

Then, using the above defined quantities, and the recursions on $m_{d,s}$, we easily find:

For $d \leq \Delta$:

$$M_{d,s} \stackrel{\Delta}{=} E(m_{d,s}) = N_{\lambda d,s} + \sum_{l \geq 3} M_{l,s} [P_{\lambda d}(l) + \frac{e^{-\lambda d}}{1-e^{-\lambda}} P_\lambda(l)] +$$

$$\begin{aligned}
& + \frac{e^{-\lambda d}}{1-e^{-\lambda}} \left[N_{\lambda,s} - P_{\lambda,s-1} \right] + U(s-2)e^{-\lambda d} \left\{ e^{-\lambda(s-2)} \sum_{m=0}^{s-2} e^{\lambda m} P_{\lambda,m} + \right. \\
& \left. + \lambda e^{-\lambda(s-1)} - \lambda \frac{e^{-\lambda}}{1-e^{-\lambda}} \right\}
\end{aligned} \tag{A.3}$$

For $d > \Delta$ and $p \leq \lfloor \frac{d-\Delta}{\Delta-1} \rfloor$, $p = 0, 1, \dots$:

$$\begin{aligned}
M_{d,s} = & N_{\lambda,\Delta,s} + \sum_{l \geq 1} M_{d-\Delta+l,s} P_{\lambda,\Delta}(l) \\
& + e^{-\lambda \Delta p} \left\{ N_{\lambda(d+p-p\Delta),s} - P_{\lambda(d+p-p\Delta),s-1} + \sum_{l \geq 3} M_{l,s} P_{\lambda(d+p-p\Delta)}(l) \right\} \\
& + \frac{e^{-\lambda(d+p)}}{1-e^{-\lambda}} \left\{ N_{\lambda,s} - P_{\lambda,s-1} + \sum_{l \geq 3} M_{l,s} P_{\lambda}(l) \right\} \\
& + U(p-1) \left\{ \frac{e^{-\lambda \Delta} (1-e^{-\lambda \Delta p})}{1-e^{-\lambda \Delta}} \left[N_{\lambda,\Delta,s} - P_{\lambda,\Delta,s-1} \right] + \sum_{1 \leq n \leq p} e^{-\lambda \Delta n} \sum_{l \geq 1} M_{d-n(\Delta-1)+l,s} P_{\lambda,\Delta}(l) \right\} \\
& + U(p-1)U(s-2)e^{-\lambda(s-1)} \sum_{m=s-1-\min(p,s-1)}^{s-2} e^{\lambda m} P_{\lambda,\Delta,m} \\
& - U(s-2)\lambda e^{-\lambda(d+p)} \left[(d+p-p\Delta)e^{\lambda p \Delta} + \frac{e^{-\lambda}}{1-e^{-\lambda}} \right] \\
& + U(s-1-p) e^{-\lambda \Delta p} P_{\lambda(d+p-p\Delta),s-1-p} \\
& + U(s-2-p) e^{-\lambda(d+s-2)} \sum_{m=0}^{s-2-p} e^{\lambda m} P_{\lambda,m}
\end{aligned} \tag{A.4}$$

Bounds

For the numbers $N_{k,s}$, we used the following bounds:

$$0 \leq N_{k,s} \leq k-1 ; \forall s \tag{A.5}$$

Regarding the numbers H_d , we used the methodology in [4], and proved that,

$$\alpha_l d + \beta_l \leq H_d \leq \alpha_u d + \beta_u , \quad d \geq 1 \quad (\text{A.6})$$

where,

$$\alpha_l = \alpha_u = [\Delta - E\{l|\Delta\}]^{-1} E\{l|\Delta\}$$

$$\beta_l = \inf_{1 \leq d \leq \Delta} Q(d) , \quad \beta_u = \max \left(-\alpha_u , \sup_{1 \leq d \leq \Delta} Q(d) \right)$$

for:

$$Q(d) = \left[\lambda d e^{-\lambda d} \right]^{-1} \left\{ E\{l|d\} + \alpha_u \left[E\{l|d\} - d - \lambda d e^{-\lambda d} \right] \right\}$$

Bounds on the numbers $M_{d,s}$ can be developed similarly with those for the numbers H_d . The former are significantly more complicated, however. Instead, we used the following simpler and intuitively clear bounds, where H_d^u denotes the upper bound on the quantity H_d :

$$0 \leq M_{d,s} \leq H_d^u \quad (\text{A.7})$$

We used the bounds in (A.7), for $d \geq 30$.

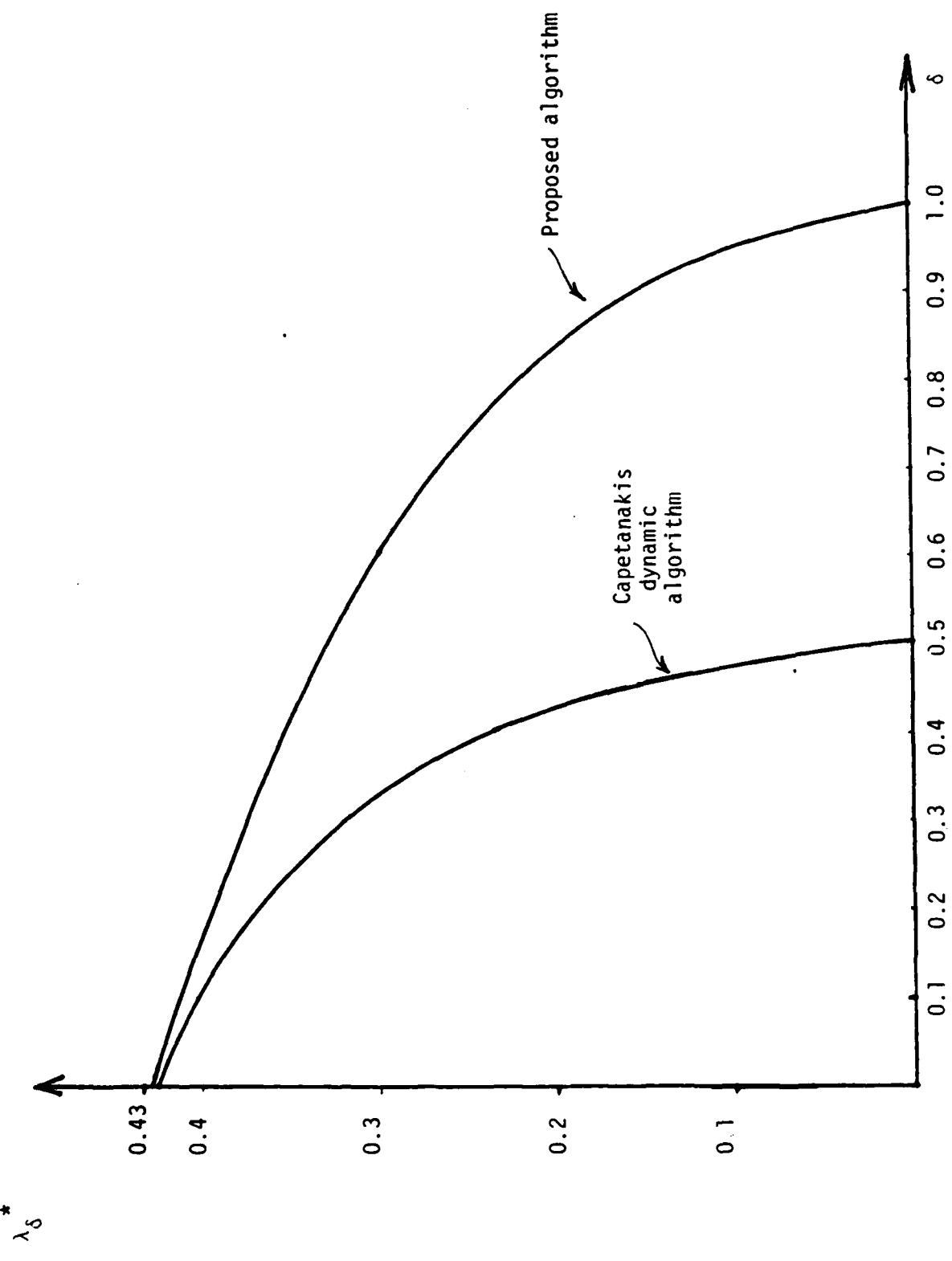


Figure 1

Throughput when an empty slot is interpreted as a collision slot, with probability δ .

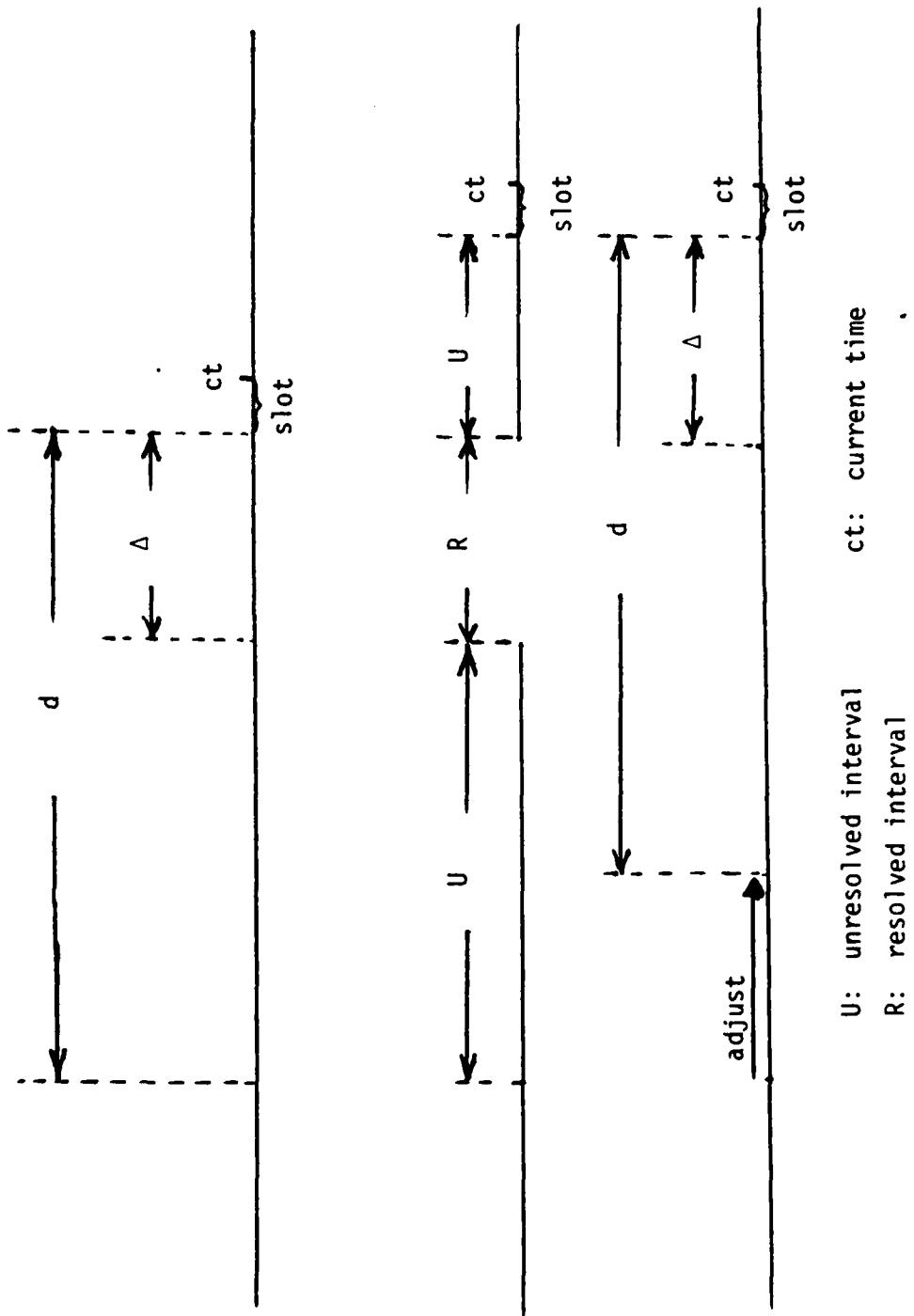


Figure 2

Window Selection in the Limited Sensing Environment

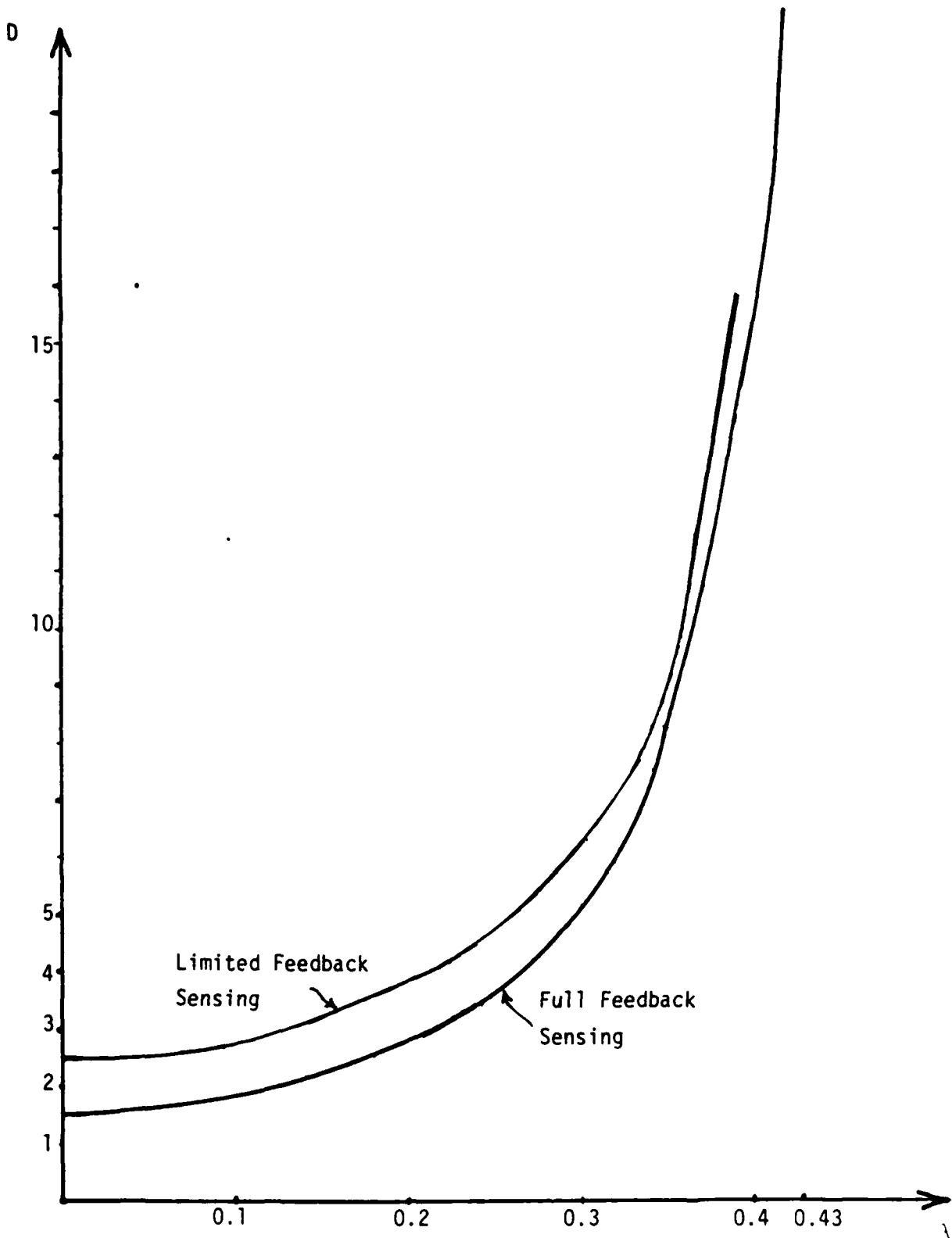


Figure 3

Expected Delays Induced by the Proposed Algorithm
in the Full Feedback Sensing and the Limited Feedback
Sensing Environments

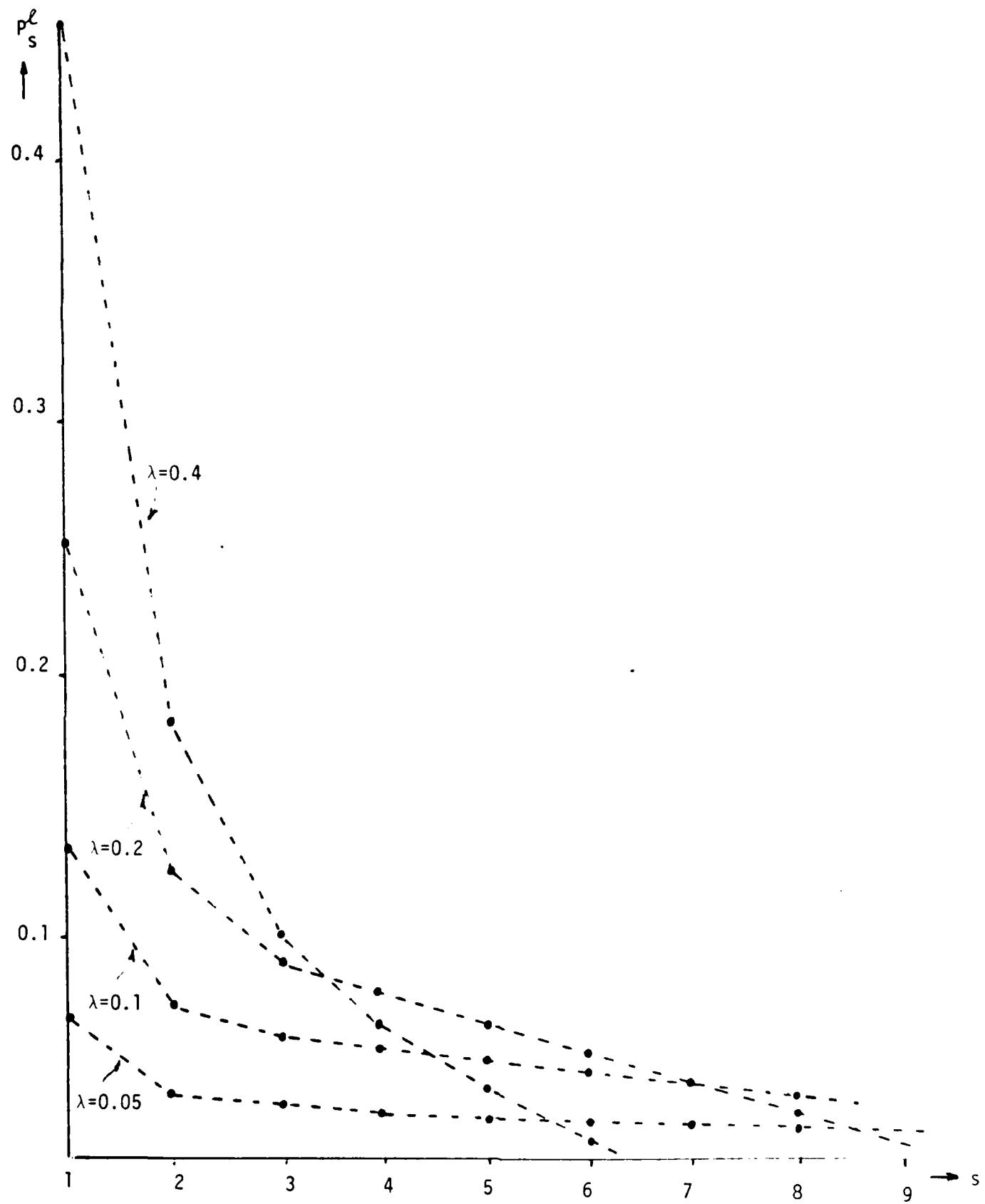


Figure 4

Lower Bounds of the Interdeparture Distribution
for Various Poisson Rates

DISTRIBUTION LIST

Copy No.

- 1 - 6 Director
 National Research Laboratory
 Washington, D.C. 20375
 Attention: Code 2627
- 7 Dr. R. N. Madan
 Code 1114SE
 Office of Naval Research
 800 N. Quincy Street
 Arlington, VA 22217-5000
- 8 Mr. James G. Smith
 Office of Naval Research
 Code 1241
 800 N. Quincy Street
 Arlington, VA 22217-5000
- 9 Professor Mike Athans
 MIT, Bldg. 35
 Cambridge, MA 02139
- 10 Professor A. Makowski
 Electrical Engineering Dept.
 University of Maryland
 College Park, MD 20742
- 11 Dr. Dave Castanon
 ALPHATECH, Inc.
 2 Burlington Executive Center
 111 Middlesex Turnpike
 Burlington, MA 01803-4901
- 12 Dr. John P. Lehoczky
 Dept. of Statistics
 Carnegie Mellon University
 Schenley Park
 Pittsburgh, PA 15213-3890
- 13 Professor Alex Levis
 MIT, Bldg. 35
 Cambridge, MA 02139
- 14 - 25 Defense Technical Information Center, 547031
 Bldg. 5, Cameron Station
 Alexandria, VA 22314

(Continued on Next Page)

26 - 27 L. Georgiadis, EE
28 - 29 M. Paterakis, EE
30 - 31 P. Kazakos, EE
32 R. J. Mattauch, EE
33 - 34 E. H. Pancake, Clark Hall
35 SEAS Publications Files
36* Office of Naval Research Resident
 Representative
 Joseph Henry Building, Room 623
 2100 Pennsylvania Avenue, N.W.
 Washington, D.C. 20037
 Attention: Mr. Michael McCracken
 Administrative Contracting Officer

*Send cover letter only

0069/ald/270R

UNIVERSITY OF VIRGINIA
School of Engineering and Applied Science

The University of Virginia's School of Engineering and Applied Science has an undergraduate enrollment of approximately 1,500 students with a graduate enrollment of approximately 560. There are 150 faculty members, a majority of whom conduct research in addition to teaching.

Research is a vital part of the educational program and interests parallel academic specialties. These range from the classical engineering disciplines of Chemical, Civil, Electrical, and Mechanical and Aerospace to newer, more specialized fields of Biomedical Engineering, Systems Engineering, Materials Science, Nuclear Engineering and Engineering Physics, Applied Mathematics and Computer Science. Within these disciplines there are well equipped laboratories for conducting highly specialized research. All departments offer the doctorate; Biomedical and Materials Science grant only graduate degrees. In addition, courses in the humanities are offered within the School.

The University of Virginia (which includes approximately 2,000 faculty and a total of full-time student enrollment of about 16,400), also offers professional degrees under the schools of Architecture, Law, Medicine, Nursing, Commerce, Business Administration, and Education. In addition, the College of Arts and Sciences houses departments of Mathematics, Physics, Chemistry and others relevant to the engineering research program. The School of Engineering and Applied Science is an integral part of this University community which provides opportunities for interdisciplinary work in pursuit of the basic goals of education, research, and public service.

END

8-87

DTIC